EL SPECTRUM Y

LOS MICRODRIVES

(C) VICENTE MENDOZA

PRODUCIDO POR VENTAMATIC
Realizado con un Spectrum
Impresora DP-80 y el
Programa CONTEXT

INSTALACION

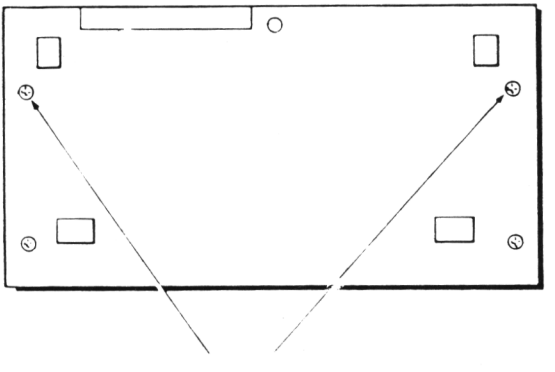
CONECTANDO EL INTERFACE 1

Al desempaquetar su Interface 1 debe encontrar:

- Un manual.
- El Interface 1 (con dos tornillos).
- Un cable plano de unos 8 cm. de largo para conectar el Interface 1 con el Microdrive.
- Un cable con un jack a cada lado para usar la red local de comunicaciones (network).

Para conectar el Interface 1 con su Spectrum realice paso a paso las siguientes operaciones:

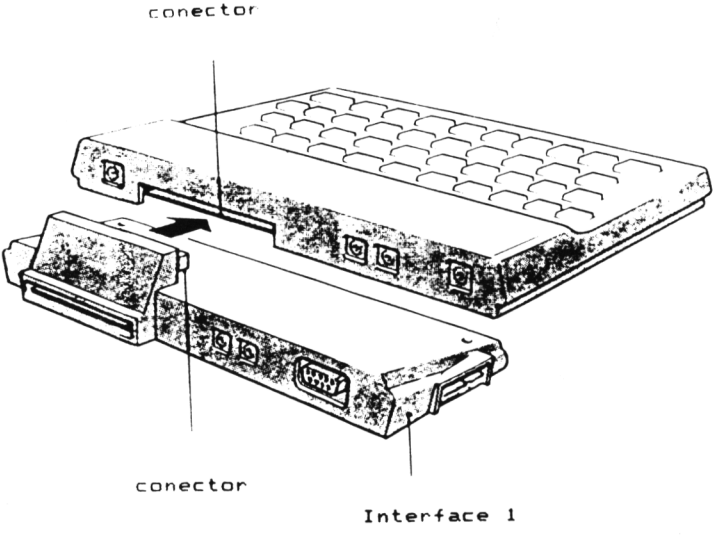
1.- Desconecte la alimentación del ordenador.
2.- Desconecte del Spectrum todos los periféricos.
3.- Con un destornillador de los llamados de estrella saque los dos tornillos laterales superiores del Spectrum, tal como muestra la figura 1. Deje los otros tornillos como están y guarde los que ha sacado yá que solo los necesitará si desea desconectar el Interface 1 en el futuro.



tornillos a sacar

4.- Conecte firmemente el Interface 1 en el conector posterior del Spectrum de modo que quede tal como se muestra en la figura 2.

5.- Fije el Interface 1 al ordenador mediante los dos tornillos largos que venian junto al Interface 1.

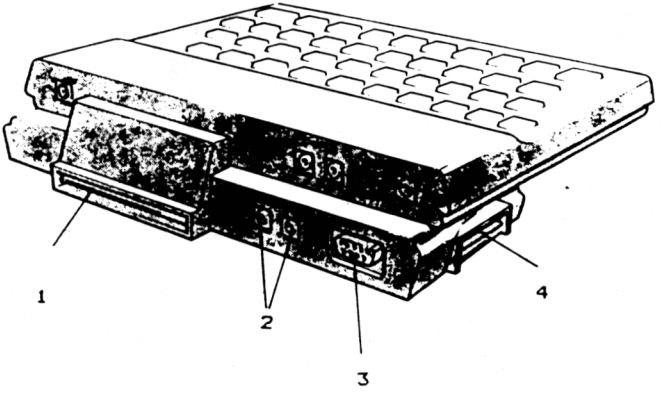


Ahora puede volver a conectar los periféricos en el conector que queda libre en el Interface 1 y conectar la alimentación al Spectrum.

LOS CONECTORES DEL INTERFACE 1

El Interface 1 tiene cuatro tipo de conectores diferentes:

- La extensión del conector del Spectrum. (1)
- Los dos conectores hembra para la red local. (2)
- El conector para el RS-232. (3)
- El conector para el cable de los Microdrives. (4)

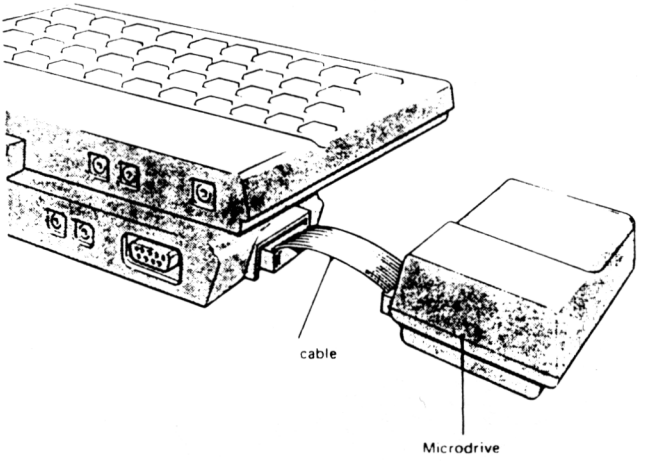


CONECTANDO EL MICRODRIVE

Al desempaquetar el Microdrive debe encontrar:

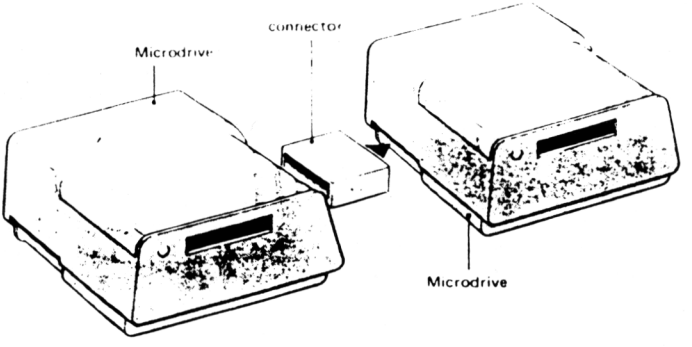
- El Microdrive.
- Un conector eléctrico de plástico rígido.
- Un cartucho de demostración.

Para conectar el Microdrive al Interface 1 inserte uno de los extremos del cable plano que venía con el Interface 1 en el conector lateral izquierdo de éste y el otro extremo en el conector lateral derecho del Microdrive. Debe quedar como muestra la figura 3.



Si desea conectar mas Microdrives, éstos deben ser conectados entre sí como sigue:

1.- Desconecte el Microdrive del Interface 1.
2.- Ponga boca abajo los Microdrives y observe que cada uno lleva atornillada una placa metálica.
3.- Desatornille estas placas y sáquelas de los Microdrives.
4.- Coloque los Microdrives boca arriba.
5.- Conecte los Microdrives entre sí por medio del conector eléctrico tal como muestra la figura 4.
6.- Con cuidado vuelva a colocar los Microdrives boca abajo y coloque una de las placas metálicas de modo que una a ambos Microdrives, es decir, con dos tornillos en cada uno de ellos.
7.- Coloque los Microdrives boca arriba y conectelos de nuevo con el Interface 1.



A partir de este momento el equipo está listo para funcionar numerándose los Microdrives del 1 al 8, contando como número 1 el más cercano al Interface 1, número 2 el siguiente, etc.

Como habrá observado el Microdrive posee en la parte frontal una abertura donde se insertarán los cartuchos y un LED (lamparita roja) que se encenderá cuando la unidad esté en marcha.

LOS CARTUCHOS

Los cartuchos vienen en una cajita protectora y deben guardarse en ella cuando no se estén usando.

Cuando saque un cartucho de su funda cuide de no tocar la cinta pues podria estropearla. Observe que tiene una marca en una de sus caras, donde podrá colocar una etiqueta para identificar el cartucho.

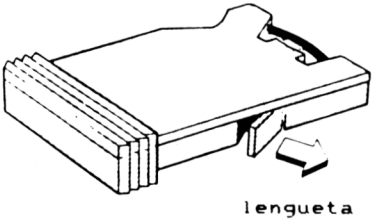
Insértelo firmemente en la ranura del Microdrive de modo que la cara con la etiqueta quede en la parte superior (a la vista mientras lo introduce). Una pequeña parte del cartucho (1/2 cm.) quedará fuera del Microdrive para posibilitar su extracción. Ahora el Microdrive y el cartucho están listos para su uso.

Cuando acabe de usarlo sáquelo con cuidado del Microdrive e introduzcalo en su funda.

NO es recomendable sacar el cartucho cuando el LED este encendido ni conectar o desconectar la alimentación con un cartucho dentro del Microdrive.

Si desea proteger un cartucho para que no pueda borrarse lo que contiene, sencillamente rompa la lengueta del lateral derecho del cartucho. Para poder volver a escribir en este, coloque un trozo de celo en el lugar de la lengueta al igual que seguramente hará con los cassettes.

Al igual que con los disquettes convencionales es muy aconsejable el tener copias de seguridad de los programas o datos y mantener los cartuchos alejados de las fuentes de magnetismo como la pantalla del TV o los altavoces de los cassettes.



lengueta

NUEVAS PALABRAS

Durante la lectura de este manual podrá comprobar que aparecen algunas palabras cuyo significado puede no quedarle demasiado claro. La razón principal puede ser que son palabras del idioma inglés o que su traducción al español no define correctamente su significado.

Estas palabras son:

ARCHIVO

En general, se habla de archivo cuando se está refiriendo a un conjunto de datos.

BUFFER

Es una zona de la memoria del ordenador que se reserva para que contenga una serie de datos.

CANAL

Se habla de canal (CHANNEL) refiriendose a un periférico ó interface.

PROGRAMA

Se trata de un conjunto de instrucciones comprensibles para el ordenador.

VIA (STREAM)

Se trata del camino que ha de abrirse para que pueda circular la información a través de él hacia un canal.

LOS COMANDOS

LOS NUEVOS COMANDOS BASIC

En el Interface 1 va alojada una nueva ROM que incorpora al Spectrum unos nuevos comandos para facilitar el uso del Microdrive y los nuevos canales del Interface 1.

Estos nuevos comandos son:

RUN

FORMAT

CAT

SAVE *, LOAD *, VERIFY * Y MERGE *

ERASE

MOVE

OPEN

INKEY

INPUT

PRINT

CLS Y CLEAR

Aunque algunos de ellos ya le serán familiares, ahora se introducen nuevas opciones de uso para ellos por lo que los veremos como si de nuevos comandos se tratara.

Antes de pasar a un estudio detallado de estos comandos puede observar como se comporta su Microdrive y probar su funcionamiento al mismo tiempo introduciendo el cartucho de demostración. Una vez introducido pulse NEW en el ordenador y despues teclee RUN y ENTER. Observará que el Microdrive se pone en funcionamiento y cargará los programas de demostración. Observe que en el borde de la pantalla aparecen intermitentes unas lineas negras. Esto indica que se esta cargando un trozo de programa. Mas adelante trataremos a fondo el modo en que estan grabados los programas o los datos en el cartucho.

R U N

Este comando adquiere una nueva dimensión al ser conectado el Interface 1 al Spectrum.

Si se ejecuta directamente despues de haber conectado la alimentación al ordenador, después de un NEW o un RANDOMIZE USR 0, se comportará del siguiente modo:

Buscara el el Microdrive 1 un programa que se llame "run", tecleado letra a letra y en minusculas.

Si lo encuentra lo ejecutará automáticamente, tal y como habrá podido comprobar con la cinta de demostración.

Una vez esté familiarizado con su Microdrive encontrará interesante el siguiente programa almacenado en cada cartucho para falicitarle la carga de cualquier programa.

Recuerde que este comando sólo actua con el cartucho que esté introducido en el Microdrive 1.

```
 10 REM CARGA DE PROGRAMAS CON MICRODRIVE
 20 CLS : INPUT "Unidad numero ";n
 30 IF n <1 OR n >8 THEN GOTO 20
 40 CAT n
 50 PRINT AT 5,10;" PULSE PARA CARGAR "
 60 PRINT AT 7,10;" 1 - Programa "
 70 PRINT AT 9,10;" 2 - Bytes "
 80 PRINT AT 11,10;" 3 - Screen$ "
 90 PRINT AT 13,10;" 4 - Matriz "
100 INPUT " Cual selecciona ? ";a
110 IF a <1 OR a >4 THEN GOTO 100
120 INPUT " Nombre ";a$
130 IF LEN a$ <1 OR a$ >10 THEN GOTO 120
140 GOTO (a*10)+140
150 LOAD*"m";n;a$: STOP
160 LOAD*"m";n;a$ CODE: STOP
170 LOAD*"m";n;a$ SCREEN$: STOP
180 LOAD*"m";n;a$ DATA a(): STOP :  REM  Cambie  a()  por  la
                                        matriz que desee cargar.
190 REM Salvar el programa
200 SAVE*"m";1;"run" LINE 10 : VERIFY *"m";1;"run"
210 STOP
```

Para salvarlo teclee GOTO 200 y luego compruebe su buén funcionamiento.

F O R M A T

Antes de poder usar un cartucho en blanco o si desea borrarlo y poder usarlo, deberá usar este comando de la siguiente forma:

FORMAT "m";1;"nom"

Donde:

"m" - le indica al Spectrum que se trata de una operación con Microdrives.

1 - Le indica al ordenador que actue sobre la unidad de Microdrives numero 1.

"nom" - Será el nombre del cartucho. Puede ser cualquier sucesión de caracteres si su longitud no es mayor de 10.

El ordenador tarda alrededor de 45 segundos en formatear un cartucho. Durante este tiempo el borde de la pantalla presentará un flash dos veces durante esta operación y luego presentará el conocido mensaje OK.

El formateado no debe ser repetido en el mismo cartucho nunca más a menos que se desee eliminar toda la información que contiene.

Durante esta ejecución el ordenador realiza tres operaciones:

- Borra toda la información del cartucho.
- Sectoriza el cartucho y verifica cada sector.
- Escribe el nombre del cartucho en cada sector de éste.

El cartucho se divide en sectores de 540 bytes, colocando una cabecera en cada sector con el nombre del cartucho y un bloque para los datos.

Compruebe que la operación se ha realizado correctamente tecleando CAT 1, que le devolverá el nombre del cartucho y el numero de Kbytes libres en éste. Si prueba a realizar la operacion FORMAT varias veces obsevará que el resultado en Kbytes libres es diferente cada vez que lo ejecuta por lo que es conveniente realizar esta operación varias veces hasta que se consigue un numero de Kbytes libres idóneo.

La capacidad mínima garantizada por Sinclair es de 85K. Si después de varios formateos no consigue este mínimo, su cartucho és defectuoso y, según dice la garantia, le será canjeado.

A pesar de ello, normalmente debe conseguir alrededor de 85 K a 90 K libres.

C A T

Este comando le devuelve el catálogo completo del contenido de un Microdrive con la siguiente información:

- El nombre del cartucho.
- Una lista de 50 programas o filas de datos no protegidos, que contiene el cartucho.
- El total de memoria que queda libre en el Microdrive, expresado en Kbytes.

La sintaxis de esta instrucción es la siguiente:

CAT n

Donde n es el número del Microdrive a catalogar.

Además, puede enviar la salida del catalogo a una via (stream), de la siguiente forma:

CAT m,n

Donde m es el número de la via (stream) y n es el número de Microdrive. Esto le permite enviar el catálogo a impresora o a un archivo de datos para ser usado de algún modo.

La forma de proteger un archivo para que su nombre no salga en pantalla al usar el comando CAT, es introducir en su nombre como primer caracter CHR$ 0 como en el siguiente ejemplo:

```
10 OPEN # 5,"m";1;CHR$ 0+"Nombre"
20 PRINT # 5; 2
30 CLOSE # 5
```

Si prueba a ejecutar el comando CAT observará que el archivo Nombre (que es el que hemos creado), no aparece por ninguna parte a pesar de que esté en el Microdrive. Para comprobar que está teclee:

```
10 OPEN # 5,"m";1;"Nombre"
20 INPUT # 5,n
30 PRINT n
40 CLOSE # 5
```

En pantalla debe aparecer el numero 2, que es el que almacenamos anteriormente en este archivo.

L O A D

A pesar de que este comando ya le debe resultar familiar su sintaxis es diferente con los Microdrives. Para cargar programas, este comando se debe introducir de esta forma:

```
LOAD *"m";1;"Nom"
```

Donde:

"m";* - Le indica al ordenador que se quiere cargar del Microdrive.

1; - Indica que unidad de Microdrive se está usando.

"Nom" - Indica el nombre del programa a cargar.

Después de una breve pausa el programa debe estar cargado.

Si se trata de cargar matrices o Bytes a continuación del comando arriba mencionado deben constar las mismas expresiones que para la carga desde el cassette es decir:

CODE - Para cargar Bytes.

DATA - Para cargar matrices de datos.

SCREEN$ - Para cargar pantallas.

Este comando también cambia un poco cuando ha de usarse con el Microdrive. La forma de hacerlo es:

```
SAVE *"M";1;"Nom"
```

Donde:

"m";* - Le indica al Spectrum que se quiere salvar en el Microdrive.
m;1; - Indica que unidad de Microdrive se está usando.
"Nom" - Indica el nombre del programa a salvar.

Después de un breve espacio de tiempo debe aparecer en pantalla el conocido mensaje OK.

Si se trata de salvar matrices, bytes, pantallas o programas con auto ejecución en una linea determinada, a continuación del comando pueden colocarse las expresiones adecuadas al igual que se hacía con el cassette, es decir:

CODE - Para salvar bytes.
DATA - Para salvar matrices de datos.
SCREEN$- Para salvar pantallas.
LINE N - Para indicar la linea (n) donde ha de empezar la ejecución del programa.

Durante la operación de salvar un programa o datos, el Microdrive realiza las siguientes maniobras:

Primero da una vuelta completa a la cinta del cartucho para comprobar que existe espacio suficiente en él para grabar el programa y que el nombre introducido es correcto y no existe otro igual en el cartucho. Después va buscando sectores libres y va grabando en cada uno de ellos 540 bytes del programa hasta completar la grabación.

Para grabar un archivo de datos y que su nombre no aparezca cuando se pide un catálogo (comando CAT), se debe incluir en su nombre el caracter con el código 0 (CHR$ 0). Por ejemplo:

```
LET G$ = CHR$ 0 + "Nombre"
```

y luego:

```
SAVE *"M";1;G$
```

VERIFY

Al ejecutar este comando el ordenador comprueba que la grabación que acabamos de hacer con el nombre "Nom" corresponde a lo que hay en la memoria del Spectrum, al igual que cuando se realiza con el cassette.

Se ejecuta de este modo:

```
VERIFY *"m";1;"Nom"
```

Donde cada una de las diferentes expresiones tiene el mismo significado que para las operaciones LOAD y SAVE y las indicaciones en el caso de bytes o pantallas son las mismas que en los casos mencionados.

Su comportamiento es similar a la operación de verificar con el cassete.

MERGE

Al ejecutar este comando el programa "Nom" se mezclará con el que esté en memoria al igual que cuando se operaba con el cassette.

Se debe ejecutar de este modo:

```
MERGE *;"m";1;"Nom"
```

Donde todas las especificaciones son las mismas que para LOAD y SAVE, por lo que no las repetiré.

ERASE

Este comando se utiliza para borrar cualquier programa o archivo de datos del Microdrive. Se utiliza según estas reglas:

ERASE "m";1;"Nom"

Donde:

m;1; - Indica la unidad de Microdrive a utilizar.
"Nom" - Indica el nombre del programa a borrar.

Observe que en este comando no se ha de indicar al Spectrum que utilice los Microdrives mediante el asterisco (*) como se hacía con LOAD, SAVE, VERIFY Y MERGE. Esto es debido a que esta instrucción es exclusiva para Microdrives por lo que el ordenador no duda en saber cual es el medio de almacenamiento en el cual se va a operar.

Si durante la operación de borrado de un programa o archivo de datos para la ejecución mediante BREAK o por otra razón el programa no estará totalmente borrado, pero sí inutilizado por lo que deberá ejecutar de nuevo el comando ERASE para completar el borrado. En esta segunda ocasión el Microdrive tardará un poco mas en borrarlo.

Resulta curioso observar el tiempo que tarda al intentar realizar el borrado de un programa que no existe. Ello da una idea de las operaciones que realiza para asegurarse de que no se está equivocando.

EJEMPLOS

Coloque un cartucho virgen en el Microdrive 1 y formateelo:

FORMAT "m";1;"Prueba 1"

Una vez aparezca el mensaje OK, pida un catalogo:

CAT 1

El cual le devolverá el nombre del cartucho y los Kbytes libres que tiene el el cartucho.

Introduzca el programa que comentamos junto al comando RUN y salvelo tal como se explica en dicho capítulo.

Luego pulse NEW y ENTER y teclee este programa:

```
10 FOR N=1 TO 100
20 PRINT N;"-";
30 NEXT N
```

Este programa imprime en pantalla los números del 1 al 100.

Salvelo mediante:

SAVE *"m";1;"Numeros"

Verifique esta operación:

VERIFY *"m";1;"Numeros"

Si el ordenador devuelve el mensaje Ok, pulse NEW y ENTER y teclee este otro programa:

```
100 FOR N=1 TO 100
110 PRINT N*N;"-" ;
120 NEXT N
```

El cual imprime los cuadrados de los números del 1 al 100.

Salvelo:

SAVE *"m";1;"Cuadrados"

Verifique:

VERIFY *"m";1;"Cuadrados"

Pulse ahora NEW y ENTER y después:

LOAD *"m";1;"Numeros"

Y se cargará el primero de los programas que hemos grabado. Teclee:

MERGE *"m";1;"Cuadrados"

y luego LIST y ENTER para comprobar que tiene en memoria los dos programas que hemos grabado. Pulse:

SAVE *"m";1;"Los dos"

y después pida un catálogo con CAT 1. Tendrá en el Microdrive los tres programas: "Numeros", "Cuadrados" y "Los dos". Borremos los dos primeros:

ERASE "m";1;"Numeros"

ERASE "m";1;"Cuadrados"

Compruebelo mediante CAT 1. Pulse NEW y RUN, para cargar el programa del capitulo en el que tratamos el comando RUN. Introduzcale la unidad 1. Luego pulse el 1 para cargar programas e introduzca el nombre "Los dos".

Con esto habremos comprobado todos los comandos que hemos visto hasta ahora. Realice Vd. todas las pruebas que crea necesarias hasta tener un buen dominio del funcionamiento del Microdrive con programas.

Si tiene alguna duda con el significado y el resultado de alguna de estas expresiones en su concepto básico, es decir, con su uso a través de cassette, consulte el manual que se le entregó con el Spectrum, ya que es muy importante tener un buen dominio de estos comandos antes de profundizar en su uso con los Microdrives.

TRATAMIENTO DE DATOS

DATOS, CANALES Y STREAMS

Como datos se entienden una serie de letras, números o simbolos, que son operados o utilizados por un programa.

Los datos pueden ser enviados o recibidos desde y hacia varias partes de las que componen su equipo (ordenador y periféricos). Cada una de estas partes suele ser llamada CANAL. Así pués los canales hacia donde la unidad central puede enviar datos pueden ser todos o algunos de los siguientes:

- a la pantalla.
- a la Impresora ZX.
- a un archivo en el Microdrive, con objeto de almacenarlos.
- a otro ordenador (Spectrum), por medio de la red local.
- a cualquier periférico (modem, impresora, etc.), por medio del Interface RS-232

Y los canales desde donde puede recibir datos pueden ser:

- el teclado.
- un archivo en el Microdrive.
- otro Spectrum, por medio de la red local.
- por el Interface RS-232, de un modem o terminal.

Los caminos que siguen estos datos tanto en un sentido como en otro deben ser abiertos por medio de las vias (streams). En el Spectrum dispone de 16 vias (streams) numeradas del 0 al 15 y el número de via (stream) debe ser precedido siempre por el simbolo .

Por si no le ha quedado claro le pondré un ejemplo. Suponga que Vd. es el ordenador y desea recibir un dato del exterior, la hora por ejemplo, de la central telefónica de su ciudad. La central telefónica sería el canal. El teléfono sería el Interface y el número que Vd. ha de marcar para conseguir el contacto sería la via (stream).

Cuatro de estas vias (streams) tiene el canal prefijado y no se pueden usar para otros canales.

Estas son:

via(stream) 0 - los datos salen en la parte inferior de la
via(stream) 1 - pantalla y entran por medio del teclado.

via(stream) 2 - los datos salen en la parte superior de la pantalla pero no pueden entrar datos por medio de ésta via(stream).

via(stream) 3 - Los datos salen por la Impresora ZX pero no pueden entrar datos por esta via(stream).

Comprobemos estas afirmaciones. Si teclea:

```
PRINT # 0;"PRUEBA": PAUSE 0
```

Obsevará que se imprime en la parte inferior de la pantalla, donde normalmente el ordenador imprime los mesajes de error. El comando PAUSE 0 evita que se borre "PRUEBA" al contestar el ordenador con el mensaje OK, hasta que se pulse una tecla. Si ahora pulsa:

```
INPUT # 0;"VARIABLE";N
```

Observará que tiene el efecto normal ya que siempre realiza la entrada de datos en esta parte inferior de la pantalla.

Compruebe ahora la via(stream) 2 mediante:

```
PRINT # 2;"Prueba 2"
```

Y el mensaje "Prueba 2" aparecerá en la parte superior de la pantalla. Compruebe también que no puede entrar datos mediante:

```
INPUT # 2;"Variable";n
```

Donde el ordenador devolverá un mensaje de error.

Todos los comandos que antes de conecta el Interface 1 producian una salida o entrada de datos en alguno de estos canales usaban estas vias(streams) automáticamente. Como hemos visto, PRINT usa la via(stream) 2 y LPRINT la via(stream) 3 automáticamente. Por lo tanto

```
PRINT "Esto es una prueba"
```

es lo mismo que

```
PRINT # 2;"Esto es una prueba"
```

Pruebelo para comprobar que tienen el mismo efecto.

Aparte de este uso de los streams establecidos Vd. puede crear los suyos propios numerándolos del 4 al 15. Como adición a estos streams puede indicarle al ordenador el periférico que desea usar por medio de los especificadores de canal. Estos son:

"K" - para el teclado.

"S" - para la pantalla.

"P" - para la Impresora ZX.

Y algunos más que estudiaremos más adelante.

Así pués K, S y P son canales establecidos. Estos requieren que Vd. use una coma (,) como separador cuando cree un stream. En el caso de otros canales puede usar una coma (,) ó un punto y coma (;).

Por ejemplo, puede abrir la via (stream) número 4 hacia la pantalla:

```
OPEN # 4,"S"
```

Y luego imprimir en ella a través de esta via (stream):

```
PRINT # 4;"Escritura en pantalla"
```

En la creación y tratamiento de canales se utilizan unos nuevos comandos. Son los siguientes:

OPEN # - Abre un stream.

INKEY # - Recoge del stream un caracter.

INPUT # - Entra en el ordenador datos del stream.

PRINT # - Saca información hacia el stream.

CLOSE # - Cierra el stream.

Veremos a continuación estos comandos más detenidamente.

Se utiliza para abrir una via(stream) de información. Se debe usar de este modo:

OPEN # x;"m";1;"Nom"

Donde:

x - Indica el número de la via(stream).
m;1; - Indica que se opera com Microdrives, unidad 1.
"Nom" - Indica el nombre de la via(stream).

Una vez ejecutada esta operación tendremos una via(stream) por la que podremos hacer circular la información y podremos estar usandola hasta que la cerremos.

Recuerde que en los casos que estamos estudiando hablamos siempre de los Microdrives, por lo que el número de la via(stream) debe ser uno entre 4 y 15. Los otros, como hemos visto, están fijados para otros periféricos.

Esta operación ha realizado lo siguiente:

- crea una nueva via(stream) y le asigna el numero x.

- pone en marcha el Microdrive y consulta su hay algún archivo con el nombre especificado. Si no lo hay, el ordenador supone que se desea crear, por lo que esperará una sentencia para introducir datos. Si el archivo existe el ordenador esperará que Vd. lo lea, porque a un archivo grabado y cerrado no se le pueden añadir más datos.

Para evitar este inconveniente, en los ejemplos que vienen al final de esta sección, le incluyo un programa que le será util.

Como yá se indicó los canales 0, 1, 2 y 3 estan prefijados por el sistema por lo que los resultados de intentar abrirlos mediante OPEN es impredecible.

PRINT

Este comando se utiliza para colocar un dato dentro de un archivo después de haber abierto una via(stream) con OPEN. Su modo de uso és:

PRINT # x;dato

Donde:

x - Indica la via(stream) de comunicación.
; - Es un separador.
dato - Es el dato a colocar en el archivo.

Dentro de las sentencias PRINT tienen una importancia vital los separadores. Como sabe, son tres:

- punto y coma (;), que no imprime nada.

- coma (,), que envia el cursor a la mitad de la linea.

- apóstrofe ('), que tiene el mismo efecto que ENTER.

Lo verá mas claro con este ejemplo:

COMANDO	ARCHIVO (# 5)
PRINT # 5;6;7	6 7
PRINT # 5;6,7	6 7
PRINT # 5;6'7	6 7

Por supuesto, como dato puede ser enviado cualquier número, caracter o variable, tal como se hace normalmente con el comando PRINT.

INPUT

Este comando toma de una via(stream) el dato o datos que se le indican. Estos datos pueden ser números o carácteres. Su modo de uso es el siguiente:

```
INPUT # x;a;b;c ...
```

Donde:

x - Indica el número de la via(stream).
a;b;c - Indica cualquier variable o conjunto de variables tanto numéricas como alfanuméricas.

Cuando se ejecuta este comando el ordenador va tomando bytes (o caracteres si se trata de cadenas alfanuméricas), hasta que encuentra un código 13. Este código 13 corresponde al retorno de carro (ENTER). Por esta razón es muy importante el modo como se crea un archivo para que luego pueda ser leído de forma sencilla y adecuada a la utilidad que se desee darle.

Debe poner atención cuando use este comando con variables alfanuméricas que contengan comillas (") dentro de la cadena pués el ordenador entenderá al llegar a este caracter que la cadena ha finalizado.

Para evitar este efecto debe usar el comando con la instrucción LINE delante de la variable. Por ejemplo:

```
INPUT # 4; LINE a$
```

INKEY$

Este comando tiene un efecto parecido al anterior pero tomando de la via(stream) un sólo carácter. Su forma de uso es:

```
INKEY$ # x
```

Y al igual que en el BASIC normal, debe ir acompañado de algún otro comando, asignándolo a una variable, imprimiéndolo, etc.

CLOSE

Cualquier via(stream) que haya sido abierta debe ser cerrada, ya que en caso contrario la información que contiene quedará mal archivada o perdida en su totalidad. La forma de hacerlo es:

CLOSE # x

Donde:

x - Es el número del archivo a cerrar.

Durante esta operación el Microdrive coloca una marca de final de archivo y salva el último sector.

Recuerde que una vez cerrado un archivo en el cual esté escribiendo ya no podrá volver a introducir más datos en él.

Por supuesto no podrá cerrar las vias(streams) 0, 1, 2 y 3 pués estan reservadas, como le indiqué, y permanecen siempre abiertas.

CLEAR

Este comando no se especifica claramente en el Manual de Sinclair. Su función es la de limpiar por completo el "area de datos de los canales", es decir, cierra todas las vias(streams) (las de 0 a 3 no las toca), y borra la información que pueda haber en los buffers de las vias(streams), al contrario que la instrucción CLOSE que los graba en el Microdrive. Su modo de uso es:

CLEAR #

Esta instrucción le permite trasladar archivos de un canal a otro por medio de las vias(streams). Su modo de uso es:

MOVE # x1 TO # x2

Donde:

x1 - Es la via(stream) de donde salen los datos.

x2 - Es la via(stream) hacia la cual se envian los datos.

Cualquiera de ambas puede ser manipulada con un Microdrive. En ese caso x1 ó x2 debe ser la expresión:

"m";1;"Nombre"

Utilizando esta expresión no es necesario abrir ningúna via(stream) con lo cual se ahorra memoria.

Las vias(streams) 1 a 3 deben ser indicadas por estos números y no se ejecutan si se utilizan los especificadores de canal: K, S ó P.

Si una via(stream) no está abierta o no encuentra el archivo en el Microdrive o existe un programa con el nombre del archivo, el ordenador devolverá el mensaje OUT OF RANGE.

Si se usa con archivos que no han sido cerrados en su creación puede haber serios problemas.

No es posible usar este comando con programas, sólo con archivos.

Algunos de los usos más útiles pueden ser:

MOVE "m";1;"Nombre" TO # 2

ó

MOVE "m";1;"Nombre" TO # 3

que mostrarán el archivo "Nombre" en pantalla y en la impresora respectivamente.

ARCHIVOS

CONSIDERACIONES SOBRE ARCHIVOS

Siempre que trabaje con archivos tenga bien presentes estas consideraciones:

- El comando PRINT# puede contener información para varias vias(streams) a la vez. Por ejemplo:

```
PRINT # 4;"tu"; # 2;"yo"; # 10;"el"
```

El cual imprimirá "tu" en la via(stream) 4, "yo" en la pantalla y "el" en la via(stream) 10.

- Si intenta crear un archivo con el mismo nombre de un programa ya existente, el ordenador devolverá el mensaje WRONG FILE TYPE pués no pueden existir dos archivos o programas con el mismo nombre.

- Algunos comandos de tratamiento de los colores de la pantalla (PAPER, INK, etc.) no tendrán efecto ejecutados detrás de una operación con vias(streams).
  El modo de evitarlo es colocando delante de estos comandos la intrucción: PRINT;. Esto evitará el problema, aunque resulte raro de entender.

- Si intenta abrir una via(stream) ya abierta el Spectrum devolverá el mensaje de error STREAM ALREADY OPEN.

- Si trata de leer un dato de un fichero que ha leído totalmente el ordenador devolverá el mensaje END OF FILE.

- Un catálogo puede ser enviado a un fichero mediante:

```
CAT # x,1
```

- Cuando se abre una via(stream), ésta no se crea realmente en el Microdrive, sino que ocupa en memoria una zona de 595 bytes con un buffer de datos de 512 bytes. Estos bytes no son enviados al Microdrive hasta que se completan los 512 o se ejecuta un comando CLOSE a ese via(stream). Es importante esta consideración, pués si se tiene en memoria un programa que la ocupe casi en su totalidad y no hay sitio para estos buffers de los vias(streams), al intentar abrir un via(stream) el ordenador puede presentar el mensaje de error OUT OF MEMORY. Además, al crearse uno de estos buffers el programa en BASIC se desplaza hacia arriba en la memoria con lo cual, si contiene código máquina en lineas REM, éste quedará inutilizado al quedar en una posición de memoria distinta de la que fué concebido para funcionar.

CLS

Este es un comando extra, que no vienen especificado en el Manual que entrega Sinclair con los Microdrives. Al usarlo, además de borrar la pantalla por completo coloca:

- la tinta (INK) en negro (0)
- el papel (PAPER) en blanco (7)
- el borde (BORDER) en blanco (7)
- el video inverso (INVERSE) en "no" (0)
- el brillo (BRIGHT) en "no" (0)
- el OVER en "no" (0)
- el FLASH en "no" (0)

O sea que realiza un borrado total de la pantalla y atributos relacionados con ella.

LIST

Este comando tampoco está en el Manual de Sinclair. Su función es enviar el listado de un programa a través de una via(stream). Resulta util si se desea el listado para usarlo con un procesador de textos o para buscar en él una palabra.

Por supuesto puede ser enviado a un archivo en el Microdrive.

Su modo de uso es:

LIST # 4;n

Donde:

4 - Es la via(stream) que hemos abierto, hacia donde va el listado.
n - Es la linea en la cual empieza el listado.

PROGRAMAS CON FICHEROS

Incluyo en este capítulo algunos programas de tratamiento de archivos que espero le sean de gran utilidad en su aprendizaje:

CREANDO UN ARCHIVO

Este programa crea un archivo con los números del 1 al 100.

```
10 REM Creacion de un archivo
20 OPEN # 5;"m";1;"Numeros"
30 FOR N=1 TO 100
40 PRINT # 5;N
50 NEXT N
60 CLOSE # 5
```

LEYENDO UN ARCHIVO

Este programa lee el archivo que hemos creado antes y lo imprime en pantalla.

```
10 REM Lectura de un archivo
20 OPEN # 5;"m";1;"Numeros"
30 FOR N=1 TO 100
40 INPUT # 5;A
50 PRINT A;
60 NEXT N
70 CLOSE # 5
```

Sin embargo este programa puede ser sustituido por una sola instrucción:

```
MOVE "m";1;"Numeros" TO # 2
```

AÑADIENDO DATOS

Como sabe, una vez cerrado un archivo no se le pueden añadir más datos. Este corto programa le introduce en el sistema para poder realizarlo.

```
10 REM Intercambio de archivos
20 OPEN # 5;"m";1;"Nombre 2"
30 MOVE "m";1;"Nombre 1" TO # 5
40 PRINT # 5;"Nuevos datos"
50 CLOSE # 5
60 ERASE "m";1;"Nombre 1"
70 OPEN # 4;"m";1;"Nombre 1"
80 MOVE "m";1;"Nombre 2" TO # 4
90 ERASE "m";1;"Nombre 2"
```

Este programa añade la frase "Nuevos datos" al archivo "Nombre 1" por medio del archivo "Nombre 2" que sirve de puente.

MEZCLANDO ARCHIVOS

Este programa mezcla dos archivos convirtiéndolos en uno solo.

```
10 REM Mezcla de archivos
20 OPEN # 5;"m";1;"Tercero"
30 MOVE "m";1;"Primero" TO # 4
40 MOVE "m";1;"Segundo" TO # 4
50 CLOSE # 5
60 ERASE "m";1;"Primero"
70 ERASE "m";1;"Segundo"
```

BUSCANDO PALABRAS

Este programa busca en un archivo del Microdrive donde se haya colocado un listado de un programa mediante el comando LIST # , una palabra en concreto.

```
10 REM Busqueda de palabras.
20 INPUT "Microdrive numero ? ";b
30 INPUT "Nombre del archivo ? ";b$
40 OPEN # 5;"m";b;b$
50 INPUT "Palabra a buscar ? "; LINE d$
60 INPUT # 4;LINE a$
70 FOR F=1 TO LEN a$ - LEN d$
80 IF a$(F TO F+LEN d$-1)=d$ THEN PRINT a$: PAUSE 50: GOTO
60
90 NEXT F
100 GOTO 60
```

Este programam finalizará con el mensaje de error END OF FILE.

LA RED LOCAL

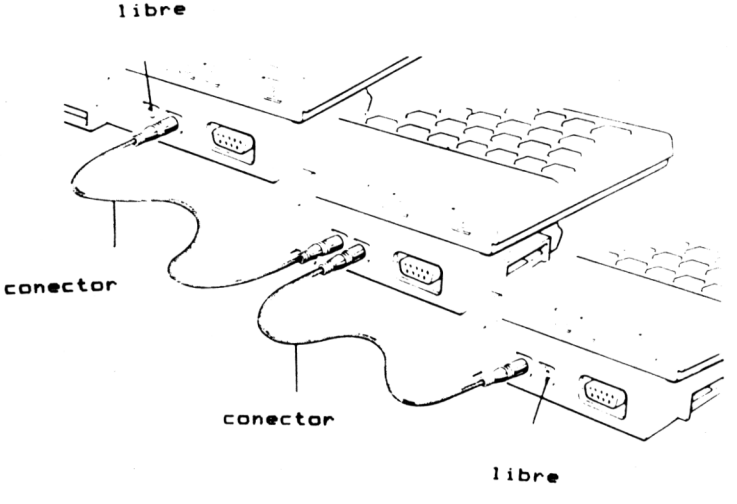
LA RED LOCAL

El area local de comunicaciones (network), permite conectar hasta 64 Spectrums y posibilita la comunicación entre ellos. A través de ella se pueden enviar tanto programas como datos, además de posibilitar el uso de un sólo Microdrive por cualquiera de estos Spectrums.

Por medio del cable que venía empaquetado con el Interface 1, puede conectar dos Spectrums entre sí del modo que se indica en la figura inferior. Nunca debe cerrar el circulo de conexiones, es decir, siempre han de quedar dos Spectrums (los de los extremos) con un conector libre.

Asegurese bien de no conectar o desconectar la alimentación de cualquier Spectrum cuando se esté realizando una comunicación, pués puede dañar al sistema.

La velocidad de transferencia por medio de esta red es de 5K por segundo, la cual es mucho más rápida que la máxima velocidad que se puede conseguir con el Interface RS-232.



Utilizando la red local puede usar vias (streams) al igual que con el Microdrive, pero con el especificador de canal "N" (o "n").

Antes de utilizarla debe asignarle a su Spectrum un número de estación de 1 a 64. Al conectar el Spectrum este número es el 1 y, si sólo se conectan dos Spectrums entre sí, no es necesario indicar el número de estación.

Para indicar este número de estación debe teclear:

FORMAT "n", número

PROGRAMAS

Por medio de la red local pueden ser enviados y recibidos programas y datos.

Para enviar un programa debe teclear:

SAVE * "n"; num

Donde:

*"n" - Indica que se desea usar la red local.

num - Indica el número de estación hacia donde se envia el programa.

Observe que usando la red local no es necesario indicar el nombre del programa.

Al igual que con los Microdrives, pueden enviarse del mismo modo matrices, pantallas y código máquina colocando tras esta expresión las conocidas DATA, SCREEN$ ó CODE según se trate.

Para recibir un programa de pulsar:

LOAD * "n"; num

Donde:

*"n" - Indica que se desea usar la red local.

num - Indica el número de estación de la cual se desea recibir.

Con este comando pueden usarse las mismas expresiones que con el anterior.

Al ejecutar estas operaciones el borde de la pantalla se activará de forma parecida (sic) a cuando se realiza SAVE.

Durante la ejecución de estos comandos el ordenador espera hasta que se ha realizado la operación completa, es decir, el que envia espera hasta que el otro reciba y viceversa.

Para verificar el programa recibido debe teclear:

VERIFY * "n"; num

y la estación que lo envió debe volver a pulsar:

SAVE * "n"; num

También puede usarse el comando MERGE, con el mismo efecto que con el cassette, del siguiente modo:

MERGE * "n"; num

Como habrá observado, al enviar o recibir programas no es necesario abrir ninguna via(stream).

DATOS

Para enviar o recibir datos por medio de la red local se usan las vias(streams) de forma parecida a como se hacía con los Microdrives y con las mismas instrucciones (OPEN, PRINT, INPUT, INKEY$, CLOSE). Para abrir una via(stream) teclee:

OPEN # x;"n"; num

Donde:

x - Indica el número de via(stream).

;"n"; - Indica que usaremos la red local.

num - Indica el número de estación a la cual enviaremos datos o de la cual los recibiremos.

Esta operación crea un buffer similar a cuando se usaba con los Microdrives pero con diferente longitud (255 bytes).

Al abrir una via(stream) debe decidir si será para enviar o para recibir datos pués una misma via(stream) no puede ser usada para ambas operaciones simultaneamente. Vd. determina a cual de ellas destinará la via(stream) al realizar la primera operación con ella. Si utiliza el comando

PRINT # x; dato

entonces será una via(stream) para enviar los datos.

Por el contrario, si en la primera operación que realice usa el comando

INPUT # x; variable (para el dato)

el ordenador entenderá que se trata de una via(stream) para recibir datos.

Igual que con los Microdrives, el comando

CLOSE # x

cierra la via(stream).

Es importante este último comando ya que el Spectrum no enviará los datos hasta que esté lleno el buffer de la via(stream) o hasta cerrarlo.

Así, si no se cierra la via(stream), la información que contenga el buffer no será enviada y la estación que la esté esperando continuará haciendolo siempre.

El efecto del comando INKEY$ # es parecido a cuando se utiliza con el teclado, es decir, devuelve la cadena vacía si en el momento de usarse no circula ninguna información por la red local.

ESTACION 0

Normalmente, enviando o recibiendo datos o programas a otra estación la red local espera hasta que la operación esté completamente realizada en ambas estaciones.

Una facilidad de la red local es la posiblidad de enviar o recibir datos sin esta espera y a todas las estaciones a la vez. A esta facilidad se accede a través de la via(stream) número 0.

Cuando se envian datos o programas a través de la via(stream) 0, se envia a todos los Spectrums conectados a la red local pero no espera la conformidad de la información. Si alguna de las estaciones está esperando de esta via(stream) recibirá la información pero las que no la esperen serán ignoradas.

Al enviar programas por esta via(stream), las estaciones receptoras deben pulsar

LOAD * "n"; 0 (y ENTER)

y la estación que envia el programa

SAVE * "n"; 0 (y ENTER)

Al enviar información a través de esta via(stream) no podrá saber que estaciones la han recibido pero en el caso del receptor podrá saber quien la envió pulsando

PRINT PEEK 23759

pués en esta posición de memoria se almacena el número de estación que envía los datos.

EL INTERFACE RS-232

EL INTERFACE RS-232

El Interface RS-232 permite al Spectrum comunicarse con multitud de periféricos e incluso otros ordenadores.

Este Interface suele incluirse como estandar en la mayoría de ellos.

Para realizar la conexión debidamente deberá consultar detenidamente el manual del periférico a conectar y hacerlo de acuerdo a estas especificaciones.

1.- No conectar.

2.- TX (entrada).

3.- RX (salida).

4.- DTR (entrada) debe ser alto cuando esté libre.

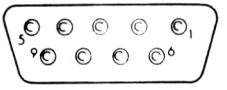
5.- CTS (salida) debe ser alto cuando esté libre.

6.- No conectar.

7.- Conexión a masa.

8.- No conectar.

9.- + 9 v.



Después debe ajustar el periférico según estos condicionantes:

- Según la aplicación, el salto automático de linea en OFF, pués el Spectrum envia en algunas ocasiones el código de salto de linea (se verá más adelante).

- La paridad en OFF.

- El número de bits, en 8.

- El número de bits de stop en 1.

En cuanto a la velocidad de transferencia (en baudios) es seleccionable desde el Spectrum (por software) y puede ser cualquiera de las estandar:

50, 110, 300, 600, 1200, 2400, 4800, 9600 y 19200

(en este caso: baudios/11 = bytes por segundo)

Lo ideal es seleccionar la máxima velocidad a la que pueda funcionar el periférico que va a conectar.

La forma de seleccionar estos baudios en el Spectrum es mediante el comando:

FORMAT "t"; baudios

ó

FORMAT "b"; baudios

"t" y "b" son, en el caso del Interface RS-232, los especificadores de canal. Ambos son diferentes y pueden usarse simultaneamente.

CANAL T

Este canal se usa para enviar a través del RS-232 cualquier texto. En relación con los códigos de los caracteres (ver apendice A del manual del Spectrum), se comporta del siguiente modo:

CODIGOS		FUNCIONAMIENTO
0-12	-	Códigos de control. Los ignora.
13	-	Retorno de carro. Envia un retorno de carro (13) y un salto de linea (10).
14-31	-	Códigos de control. Los ignora.
32-127	-	Códigos ASCII. Los envia sin modificar.
128-164	-	Códigos gráficos. Envia un interrogante (?).
165-255	-	Comandos del Spectrum. Los envia caracter a caracter.

Al ejecutar un comando INPUT o INKEY$ por este canal el Spectrum espera 7 bits, ignorando el octavo.

Este canal es idóneo para el uso de impresoras con Interface RS-232 pués una vez abierto un stream mediante:

```
OPEN # x; "t"
```

los comandos LLIST y LPRINT se ejecutan directamente hacia este canal.

Como ha visto, pueden abrirse streams en el canal "t" y usar todos los comandos que usábamos con archivos, es decir:

```
OPEN # x; "t"

PRINT # x; dato

INPUT # x; dato

INKEY$ # x

CLOSE # X
```

Pero si tiene que enviar un código de control a una impresora o necesita que el Spectrum tenga en cuenta los 8 bits que entran por el RS-232, tendrá que usar otro canal para ello: el "b" (binario).

CANAL B

Como hemos dicho, este canal permite enviar o recibir cualquier código de 8 bits, pués no realiza ningún tipo de conversión de códigos. Además este canal le permite enviar o recibir programas por medio del RS-232.

Para ello debe usar primero el comando:

FORMAT "b"; baudios

Luego si quiere enviar un programa pulse:

SAVE * "b"

y para recibirlo:

LOAD * "b"

Las demás expresiones usuales (SCREEN$, CODE, LINE), están permitidas del siguiente modo:

SAVE * "b"; SCREEN$

ó

SAVE * "b"; LINE n

Como en el canal "t", pueden ser utilizados los comandos:

OPEN # x; "b"

PRINT # x; dato

INPUT # x; dato

INKEY$ # x

CLOSE # x

CONSIDERACIONES

- No es posible conectar dos periféricos a la vez via RS-232 pero sí que diferentes canales envien o reciban información a través de él. Tampoco es posible que diferentes canales tengan diferente velocidad en baudios.

- Al seleccionar una velocidad en baudios, esta se ajustará al número inferior más próximo de los estandar, es decir, si por ejemplo intenta utilizar una velocidad de 70 baudios, el Spectrum la ajustará a 50 por ser la más próxima dentro de las estandar.

- Cuando envia datos via RS-232 el borde de la pantalla se vuelve negro si el periférico da al Spectrum la señal de ocupado y el ordenador espera la señal de libre para enviar la información.

- Cuando cierra (CLOSE) una via(stream) que operaba con el RS-232, el Spectrum envia el código 13 hacia ésta. Esto no se efectua si utiliza el comando CLEAR .

- Si olvida la velocidad en baudios que está usando, puede conocerla mediante:

  PRINT INT (3500000/((PEEK 23747+256*PEEK 23748+2)*26))

EJEMPLOS DE USO DEL RS-232

Supongamos que quiere tener por impresora un listado doble alto, que el código doble alto de su impresora es el 14 y funciona a una velocidad de 1200 baudios. Para poder hacerlo teclee este programa:

```
10 REM Listado doble alto
20 FORMAT "t"; 1200
30 FORMAT "b"; 1200
40 OPEN # 5; "t"
50 OPEN # 10; "b"
60 PRINT # 10; CHR$ 14;
70 LIST # 5
80 CLOSE # 5
90 CLOSE # 10
```

Se puede obtener una copia de la pantalla por impresora mediante este programa (en baja resolución):

```
 10 REM Copy
 20 FORMAT "t"; baudios
 30 OPEN # 5;"t"
 40 FOR N=0 TO 21
 50 FOR M=0 TO 31
 60 LET B$= SCREEN$ (N,M)
 70 IF B$="" THEN LET B$=" "
 80 PRINT # 5;B$;
 90 NEXT M
100 PRINT # 5
110 NEXT N
```

Otra de las ventajas que incorpora el RS-232 si lo usa con una impresora es la posibilidad de usar los comandos LLIST y LPRINT directamente.

Para ello habrá de formatear primero el canal "t" a la velocidad en baudios de su impresora y luego abrir la via(stream) número 3 (SIEMPRE LA 3) hacia ese canal, o sea:

```
FORMAT "t"; baudios        (ENTER)

OPEN # 3;"t"               (ENTER)
```

y el sistema estará listo para aceptar estos comandos:

```
LLIST

LPRINT
```

LA MEMORIA

ESTADO DE LA MEMORIA

Si ha leido antentamente las páginas anteriores habrá observado que a partir de la conexión del Interface 1 al Spectrum la memoria de éste ha sufrido grandes modificaciones.

La más destacable se indicó en un capitulo anterior y consiste en los movimientos arriba y abajo de la memoria del programa BASIC cuando se usa el Microdrive. Como apuntamos en varios capitulos ello se debe a los buffers que necesitan los canales y a que, cada vez que se usa el Microdrive, el Spectrum se crea en memoria un mapa completo de él, sin embargo, el usuario no notará nada anómalo si el programa que ejecuta es BASIC, sin POKES ó PEEKS en la propia zona de programa (por ejemplo, datos ó código máquina en líneas REM).

Para evitarle problemas con la memoria se incluyo un programa que le informará del espacio que ocupa cada cosa y la memoria que tiene libre en cualquier momento. Si desea usarlo de la mejor manera, cambie los números de linea e introduzcalo como una subrutina dentro de su propio programa y acceda a ella cada vez que lo crea necesario.

```
  1 REM  ESTADO DE LA MEMORIA
  2 REM
  3 REM
  4 REM   § VICENTE MENDOZA
  5 REM
  6 REM
  7 REM
  8 REM
  9 REM
 10 DEF FN m(n)=PEEK n+256*PEEK
 (n+1)
 20 PAPER 7: INK 0: BORDER 7: C
LS
 30 DRAW 255,0: DRAW 0,175: DRA
W -255,0: DRAW 0,-175
 40 PRINT AT 1,5; PAPER 2; INK
6;"ESTADO DE LA MEMORIA"
 50 PRINT AT 4,1;"RAM (total)";
AT 4,18;(FN m(23732)-16383)/1024
;AT 4,23;" Kbytes"
 60 PRINT AT 6,1;"Mapas MICRODR
IVE";AT 6,18;FN m(23631)-23734;A
T 6,23;" bytes"
 70 PRINT AT 8,1;"Programa BASI
C";AT 8,18;FN m(23627)-FN m(2363
5);AT 8,23;" bytes"
 80 PRINT AT 10,1;"Variables";A
T 10,18;FN m(23641)-FN m(23627)-
1;AT 10,23;" bytes"
 90 PRINT AT 12,1;"Memoria libr
e";AT 12,18;FN m(23730)-FN m(236
41);AT 12,23;" bytes"
100 PRINT AT 16,1;"Graficos def
inibles:"
110 REM Las mayusculas de la
        linea 120 deben ser
        introducidas en modo
        grafico.
        (CAPS SHIFT+9)
120 PRINT AT 18,5;"ABCDEFGHIJKL
MNOPQRSTU"
```

El programa le presentará en pantalla:

```
     ESTADO DE LA MEMORIA

RAM (total)        47.00 Kbytes
Mapas MICRODRIVE 0       bytes
Programa BASIC     1148  bytes
Variables          0     bytes
Memoria libre      39444 bytes

Graficos definibles:
    ABCDEFGHIJKLMNOPQRSTU
```

Veamos ahora como conseguir acceder al mapa del Microdrive que se crea en memoria cuando lo usamos.

Vamos a crear un archivo en el Microdrive donde tendremos los números del 1 al 2000 mediante este programa:

```
10 REM Numeros
20 OPEN # 5;"m";1;"Numeros"
30 FOR N=1 TO 2000
40 PRINT # 5; N;
50 NEXT N
60 CLOSE # 5
```

Una vez ejecutado el programa pulse NEW y teclee este programa:

```
10 REM Registros-Sectores
20 OPEN#5;"m";1;"Numeros"
30 PRINT "Registro ";PEEK 23912;" en el Sector ";PEEK 23885
40 POKE 23856,3
50 GOTO 30
```

Este programa se detendrá con el mensaje de error END OF FILE.

Explicaré ahora que es lo que realiza el programa.

En primer lugar ha debido pulsar NEW con lo cual se borra cualquier contenido de la memoria incluídas las vias(streams) que pudiera haber. Así el buffer de la primera via(stream) que se utilice se creará a partir de la dirección de memoria 23844.

Esto es necesario para poder calcular los datos de esta via(stream), pués de no hacerlo hubieramos tenido los datos de la primera via(stream) que hubiera en memoria.

Luego hemos abierto una via(stream) para la lectura de los datos. Al realizar esta operación, dentro del buffer se han creado una serie de variables, que hemos consultado con PEEK 23912 Y PEEK 23885.

Veamos más a fondo el contenido de estos buffers.

LOS BUFFERS DE LOS CANALES DEL MICRODRIVE

Cada vez que abre una via(stream) con el comando OPEN se crea en memoria un fichero (llamado CHANNEL) dentro de la zona llamada CHANS en el manual de Sinclair (capitulo 24). Este fichero tiene una longitud de 595 bytes de los cuales 512 son un buffer donde se almacena la información que maneja la via(stream).

El resto de bytes los ocupan unas variables que contienen información sobre la via(stream). Estas variables son las siguientes:

BYTES	NOMBRE	CONTENIDO
0-1		Dirección 8. (0008h). dirección de la rutina en INPUT
2-3		Dirección 8. (0008h). dirección de la rutina en OUTPUT.
4		Caracter "M".
5-6		Dirección de la subrutina de salida en ROM "oculta".

BYTES	NOMBRE	CONTENIDO
7-8		Dirección de la subrrutina de entrada en la ROM "oculta".
9-10		Contiene el número "595". (Longitud del buffer de la via(stream)).
11-12	CHBYTE	Contiene el número del siguiente byte a añadir ó a leer del buffer de datos en el rango de 0 a 512 inclusive.
13	CHREC	Contiene el número de registro con el que está trabajando, dentro del rango de 0 a 255.
14-23	CHNAME	Contiene los 10 bytes que forman el nombre de la via(stream).
24	CHFLAG	Contiene un señalizador de lectura/ escritura, es decir el bit 0 está a 1 si la via(stream) está abierta para escritura y está a 0, cuando la via(stream) se usa para lectura de datos. Los bits 1 a 7 no se usan y están todos a 1.
25	CHDRIV	Contiene el número de Microdrive que se está usando, dentro del rango de 1 a 8.
26-27	CHMAP	Dirección del MAPA de este Microdrive.
28-39	CHMAP	Estos 27 bytes son la zona de trabajo donde se va elaborando una cabecera que se grabará en el Microdrive conteniendo un resumen de los datos del archivo (nombre, logitud, etc.).
40	HDFLAG	Contiene un señalizador que usa sólo el bit 0 para señalizar la cabecera.
41	HDNUMB	Contiene el número del sector que se ha grabado o leido.
42-43		NO USADOS.
44-53	HDNAME	Contiene el nombre del cartucho que se está usando.
54	HDCHK	Sumatorio de comprobación de la cabecera.
55-66		12 bytes conteniendo el preambulo del bloque de datos. Marca el comienzo de la zona de manejo de los datos.

BYTES	NOMBRE	CONTENIDO
67	RECFLG	Contiene un indicador: bit 0 a 0 - no es cabecera. bit 1 a 0 - no es final del fichero. bit 1 a 1 - es final del fichero. bit 2 a 0 - es archivo de escritura. bit 2 a 1 - no es archivo de escritura.
68	RECNUM	Numero de registro dentro del rango de 0 a 255.
69-70	RECLEN	Contiene un número que indica la longitud del registro dentro del rango de 0 a 512.
71-80	RECNAM	Contiene los caracteres que forman el nombre del archivo.
81	DESCHK	Contiene un sumatorio de los 14 bytes anteriores.
82-593	CHDATA	Contiene el buffer con los 512 bytes de datos.
594	DCHK	Contiene un sumatorio para la comprobación de los 512 bytes anteriores (datos).

MAPAS DEL MICRODRIVE

Como habrá observado, en la explicación de algunas de estas variables se habla de los mapas de los Microdrives. Estos se crean en memoria para que el ordenador conozca cuales y cuantos sectores estan libres, ocupados o inutilizados.

En principio el cartucho del Microdrive contiene 180 sectores aunque el software del sistema (ROM) está preparado para manejar hasta 256 sectores.

Por cada via(stream) que abra para usar con el Microdrive se crea en la memoria un mapa del cartucho conteniendo los sectores libres y lo que no se pueden usar. Vamos a echar una mirada a estos mapas para comprender mejor el funcionamiento del Microdrive.

El mapa contiene un total de 32 bytes. Cada uno de estos bytes contiene el estado de 8 sectores del Microdrive (uno por cada bit).

Si el sector contiene datos o es inutilizable el bit será 1 y si está libre el bit será 0.

El siguiente programa le mostrará el mapa completo de un cartucho.

Para que funcione correctamente puede primero formatear el cartucho mediante:

```
FORMAT "m";1;"cartucho 1"
```

Debe pulsar NEW seguido de ENTER para eliminar cualquier via(stream) que hubiera en memoria.

Luego teclee este programa:

```
10 REM Mapa del Microdrive
20 OPEN # 5;"m";1;"mapa"
30 FOR N=1 TO 32
40 LET Y= PEEK (23792+N)
50 FOR X=1 TO 8
60 PRINT Y/2 <> INT (Y/2)
70 LET Y=INT (Y/2)
80 NEXT X
90 NEXT N
```

Y hagalo funcionar mediante RUN.

Este programa abre una via(stream) para que se cree en memoria el mapa del Microdrive y luego consulta los 32 bytes que la componen bit a bit, imprimiendo en pantalla un 0 si el sector está libre ó un 1 si está inutilizado.

Presentará en la pantalla algo parecido a esto:

```
01000000000000000000000000000000
00000000000000000000000000000000
00000000000000000000000000000000
00000000000000000000000000000000
00000000000000000000000000000000
00000000000000000000111111111111
11111111111111111111111111111111
11111111111111111111111111111111
```

Donde puede observar que el sector 1 está inutilizado así como los sectores del 181 al 256, con lo cual quedan 199 sectores libres que dan un total de capacidad de 90 Kbytes (512 bytes por sector).

LOS CANALES DE LA RED LOCAL

Al igual que con los Microdrives, cuando se abre una via(stream) direccionada a la red local se crea en memoria un buffer (zona de memoria para trabajo del ordenador), conteniendo los datos a enviar o recibir y una serie de variables que usa el ordenador.

Esta área de información tiene una longitud de 276 bytes y contiene un buffer de datos de 255 bytes además de estas variables:

BYTE	NOMBRE	CONTENIDO
0-1		Dirección 8. (0008h)
2-3		Dirección 8. (0008h)
4		Contiene el caracter "N".
5-6		Dirección de la subrrutina de salida en ROM.
7-8		Dirección de la subrrutina de entrada en ROM.
9-10		Contiene el número "256". (Longitud del buffer de la via(stream)).
11	NCIRIS	Contiene el número de la estación de destino.
12	NCSELF	Contiene el número de estación propio.
13-14	NCNUMB	Número de bloque, dentro del rango de 0 a 65535.
15	NCTYPE	Contiene un código para identificar el tipo de dato. Es 0 si son datos y 1 si ha de dar el mensaje de error END OF FILE.
16	NCOBL	Contiene la longitud del buffer de datos a enviar. Contendrá un 0 cuando se esté recibiendo.
17	NCDCS	Sumatorio de comprobación para los datos.
18	NCHCS	Sumatorio de comprobación de la cabacera.
19	NCCUR	Posición del último caracter recibido en el buffer.

BYTE	NOMBRE	CONTENIDO
20	NCIBL	Contiene el número de bytes que deben ser leidos del buffer de datos.
21-275	NCB	Contiene el buffer de datos. (Logitud: 255 bytes)

LAS VARIABLES DEL SISTEMA

Además de las variables del sistema que se crean al usar el Microdrive o la red local en las vias(streams) de éstos, con el Interface 1 se añaden más variables del sistema a las que constan en el Capítulo 25 del manual de Sinclair.

Estas nuevas variables son:

DIRECCION	NOMBRE	CONTENIDO
23734	FLAGS 3	Indicadores. Si los bits están a 1 indican lo siguiente: bit 0-Cuando se usa la Extensión del BASIC que contiene el Interface 1. bit 1-Durante la ejecución del comando CLEAR . bit 2-Cuando la variable ERRSP (23613), está alterada. bit 3-Cuando está funcionando la red local. bit 4-Cuando se ejecuta el comando LOAD. bit 5-Cuando se ejecuta el comando SAVE. bit 6-Cuando se ejecuta el comando MERGE. bit 7-Cuando se ejecuta el comando VERIFY.

DIRECCION	NOMBRE	CONTENIDO
23735	VECTOR	Contiene la dirección de salto cuando la ROM del Interface 1 dá un error. Normalmente: 23735 contiene: 240 23736 contiene: 1

Se utiliza para ampliar el juego de instrucciones por el usuario.

23737	SBRT	Rutina usada por la nueva ROM para llamar a las subrutinas de la ROM del Spectrum, en la forma:
		LD HL, valor
		CALL subrutina
		LD (5CBAH), HL
		RET

23747	BAUD	Contiene un número de dos bytes que representan la velocidad en baudios del RS-232.

La formula usada para el cálculo de esta velocidad es:

baudios = (3500000 / (26 * velocidad en baudios)) - 2

Puede usarse esta dirección para hacer que el Spectrum funcione a una velocidad en baudios diferente a las indicadas como estandar.

23749	NTSTAT	Contiene el número de estación de la red local que le corresponde al propio Spectrum, dentro del rango de 1 a 64.

23750	IOBORD	Contiene el número del color del borde de la pantalla durante las operaciones de entrada y salida.

Si realiza un POKE en esta dirección entre 0 y 7 el borde cambiará al color que le indique.

Si no desea el efecto de flash durante las operaciones use:

POKE 23750, INT (PEEK 23624/8)

DIRECCION	NOMBRE	CONTENIDO
23751	SER_FL	Contiene dos bytes que corresponden a la zona de trabajo del RS-232. El primero son indicadores y el segundo es el caracter recibido.
23753	SECTOR	Espacio de trabajo usado por los Microdrives, normalmente para contar los sectores del cartucho.
23755	CHADD_	Posición de almacenamiento temporal de la variable CH_ADD, mientras chequea la sintaxis de los nuevos comandos en una linea.
23757	NTRESP	Almacenamiento del código de respuesta de la red local.
23758	NTDEST	Principio de buffer de la red local. Contiene el número de la estación de destino.
23759	NTSRCE	Contiene el número de la estación fuente.
23760	NTNUMB	Número de bloque de la red local, dentro del rango de 0 a 65535.
23762	NTTYPE	Código del tipo de cabecera de la red local. Contiene un 0 si son datos ó 1 si ha llegado al final del archivo.

DIRECCION	NOMBRE	CONTENIDO
23763	NETLEN	Contiene la longitud del bloque de datos de la red local dentro del rango de 0 a 255.
23764	NTDCS	Sumatorio de comprobación del bloque de datos de la red local.
23765	NTHCS	Sumatorio de comprobación de la cabecera de la red local.
23766	D_STR1	Principio de los primeros 8 bytes especifcadores de canal. Estos bytes indican el número de unidad de Microdrive o el número de la estación de destino de la red local o la velocidad en baudios del RS-232.
23768	S_STR1	Número de via(stream), dentro del rango de 0 a 15.
23769	L_STR1	Especificador de canal. ("M", "N", "T" ó "B")
23770	N_STR1	Longitud del nombre del archivo.
23772	T_STR1	Principio del nombre del archivo. Normalmente en el espacio de trabajo.
23774	D_STR2	Principio de los segundos 8 bytes especificadores de archivo.

DIRECCION	NOMBRE	CONTENIDO
23782	HD_00	Principio de la zona de trabajo para SAVE, LOAD, VERIFY Y MERGE. Contiene el código del tipo de archivo. Los códigos son: 0-Si es un programa. 1-Si es una matriz numérica. 2-Si es una matriz alfanumérica. 3-Si son bytes.
23783	HD_0B	Longitud de los datos, dentro del rango de 0 a 65535.
23785	HDD	Principio de los datos, dentro del rango de 0 a 65535.
23787	HD_0F	Longitud del programa, dentro del rango de 0 a 65535.(Puede contener el nombre de la matriz a cargar). Usada también por el código de llamada "32H".
23789	HD_11	Número de linea donde comienza la auto - ejecución de un programa.
23791	COPIES	Número de copias a realizar por el comando SAVE. Se coloca a 1 después de ejecutado.

Alterando el contenido de esta posición de memoria puede hacer que automáticamente se realicen varias copias del mismo programa en un cartucho del Microdrive, con lo que verá reducido considerablemente el tiempo de acceso a él, aunque obviamente el programa ocupará más memoria en el cartucho.

Para ello pulse antes del comando SAVE:

```
POKE 23791, NUM
```

Donde NUM es el número de copias a realizar.

Para eliminar un programa grabado n veces con POKE 23791,n: SAVE *"m";1;"prog", hay que hacer n veces ERASE "m";1;"prog".

MENSAJES DE ERROR

MENSAJES DE ERROR

El Interface 1 incorpora nuevos mensajes de error a los indicados en el Apendice B del manual de Sinclair. Estos nuevos mensajes, como es lógico se refieren a las nuevas instrucciones. Son los siguientes:

CODE ERROR

Ha intentado cargar un bloque de bytes más largo de lo que Vd. especificó en el comando LOAD.

DRIVE "WRITE" PROTECTED

Ha intentado escribir en un cartucho que estaba protegido (por haber quitado la lengueta de protección).

FILE NOT FOUND

Ha intentado cargar un archivo o programa que no existe o parte de él no puede ser encontrado. Puede ser por no haber cerrado un archivo o porque haya sido dañada al desconectar la alimentación con el cartucho en el Microdrive.

INVALID DEVICE EXPRESION

Ha usado un especificador de canal distinto de s, p, k, m, n, t ó b. También se produce cuando haya colocado un punto y coma en vez de una coma usando los canales s, p ó k.

INVALID DRIVE NUMBER

Ha especificado un número de Microdrive fuera del rango de 1 a 8 .

INVALID NAME

Ha intentado salvar un programa o crear un fichero cuyo nombre es una cadena vacía o una cadena con más de diez caracteres.

INVALID STATION NUMBER

Ha intentado especificar un número de estación en la red local fuera del rango de 0 a 64, ó fuera del rango de 1 a 64 si le ocurre al usar el comando FORMAT.

INVALID STREAM NUMBER

Ha intentado especificar una via(stream) con un número fuera del rango de 0 a 15.

MERGE ERROR

Ha intentado usar el comando MERGE con una matriz o con un bloque de bytes. También ocurre al intentar utilizarlo con un programa salvado mediante SAVE ... LINE ....

MICRODRIVE FULL

Ha intentado escribir en un cartucho donde no había espacio libre. Si se trata de un programa puede volver a salvarlo usando un cartucho vacío. Si se trataba de un archivo de escritura deberá borrarlo por completo del cartucho al no haber espacio para colocar el sector con la marca de fin de archivo.

MICRODRIVE NOT PRESENT

Ha intentado usar un Microdrive que no está conectado al ordenador ó un Microdrive que no contiene ningún cartucho.

MISSING BAUD RATE

No ha especificado la velocidad en baudios.

MISSING DRIVE NUMBER

No ha especificado el número de Microdrive.

MISSING NAME

Ha olvidado indicar el nombre del programa o archivo.

MISSING STATION NUMBER

Ha olvidado especificar el número de estación de la red local.

PROGRAM FINISHED

Ha intentado ejecutar un programa a partir de una linea mayor que la última o ha usado el comando RUN sin programa en memoria.

READING A "WRITE" FILE

Ha intentado leer datos de un archivo no existente o abierto para escritura.

STREAM ALREADY OPEN

Ha intentado abrir una via(stream) para un nuevo canal que estaba abierta. Sólo puede abrir una via(stream) si previamente la ha cerrado.

VERIFICATION HAS FAILED

Lo que ha intentado verificar no corresponde con lo que hay en memoria.

WRITING TO A "READ" FILE

Ha intentado escribir en un archivo que ya existia. Primero debe borrar el archivo existente o crear otro nuevo.

WRONG FILE TYPE

Ha intentado usar los comandos SAVE ó MOVE con un archivo no abierto. También ocurre si intenta usar el comando MERGE con un archivo de escritura y cuando intenta usar el comando LOAD con bytes o matrices como si se trataran de programas.

HOOCK CODE ERROR

Este código de error no está especificado en el manual de Sinclair y su explicación es un tanto complicada. Aparece cuando el sistema intenta acceder a la ROM del Interface 1 por una dirección por la que no debe ser accedida. Es un mensaje de error que aparece en muy pocas ocasiones.

RESUMEN

RESUMEN DE LOS COMANDOS

Resumen de los nuevos comandos que se incorporan al Spectrum con el Interface 1 y el Microdrive.

En este resumen:

n - Indica el número de Microdrive o de la estación de la red local.

x - Indica un número de via(stream).

v - Indica una variable de cualquier tipo.

canal - Indica una expresión dirigida a un canal (;"m";1;....).

CAT n

Presenta en pantalla una lista en orden alfabético de hasta 50 programas ó archivos que contiene el cartucho insertado en el Microdrive junto con el nombre de éste y los Kbytes que quedan libres en él.

CAT # x; n

Envía el catálogo del Microdrive n hacia la via(stream) x.

CLOSE # x

Cierra la via(stream) x. Si hay datos en el buffer de la via(stream) los envía antes de cerrarlo.

ERASE "m";n;"nom"

Borra el programa o archivo "nom" del cartucho insertado en el Microdrive n.

FORMAT "m";n;"nom"

Formatea un cartucho insertado en el Microdrive n, con el nombre "nom".

FORMAT "n";n

Coloca el número de estación propio de la red local a n.

FORMAT "t"; n

Coloca la velocidad en baudios (n) para usar en el Interface RS-232 el canal de texto ("t").

FORMAT "b"; n

Igual que el anterior pero para usar el canal de binario ("b"). Ambas velocidades en baudios deben ser iguales.

INKEY$ # X

Toma un caracter de la via(stream) como una cadena de un sólo caracter. Si no encuentra ningún caracter en el momento de realizar la exploración de la via(stream) asigna la cadena vacía.

INPUT # x; v

Toma información de la via(stream) para la variable "v" hasta encontrar un retorno de carro. La via(stream) debe haber sido abierta previamente para entrada de datos. Si coloca un texto o usa la coma (,) como separadora en el comando INPUT #, éste será enviado hacia la via(stream). Puede usarse el comando LINE como se hacía habitualmente.

LOAD * canal

Carga un programa, matriz de datos o un bloque de bytes del canal especificado. Sólo pueden usarse los canales "m", "n", y "b". Al ejecutar este comando se borrará el programa que haya en memoria para cargar éste.

MERGE * canal

Tiene el mismo efecto que LOAD * pero sin borrar el programa que hay en la memoria, al igual que con el cassete.

MOVE fuente TO destino

Traslada un archivo de "fuente" a "destino". Fuente y destino son vias(streams) o canales. Acaba su ejecución cuando encuentra la marca de fin de archivo en el "fuente".

OPEN # via, canal

Asigna la via(stream) al canal especificado.

PRINT # via ....

Envia los datos (....) hacia la via(stream) indicada. La via(stream) debe haber sido abierta (OPEN) previamente para salida de datos.

SAVE * canal

Salva un programa, matriz de datos o un bloque de bytes en el canal especificado. Sólo pueden usarse los canales "m", "n" y "b".

VERIFY * canal

Compara el programa, matriz o bloque de bytes, que deben entrar al igual que con LOAD, con lo que hay en la memoria del ordenador.

LAS TECLAS DE LOS NUEVOS COMANDOS

A modo de orientación y hasta que coja un poco de práctica le incluyo un listado de las teclas que ha pulsar para conseguir los nuevos comandos.

COMANDO	MODO	TECLA
CAT	E	SYMBOL SHIFT - 9
CLOSE #	E	SYMBOL SHIFT - 5
ERASE	E	SYMBOL SHIFT - 7
FORMAT	E	SYMBOL SHIFT - 0
MOVE	E	SYMBOL SHIFT - 6
OPEN #	E	SYMBOL SHIFT - 4

El resto de comandos son compuestos de un comando que yá se usaba anteriormente y un signo por lo que requieren pulsar dos o más teclas.